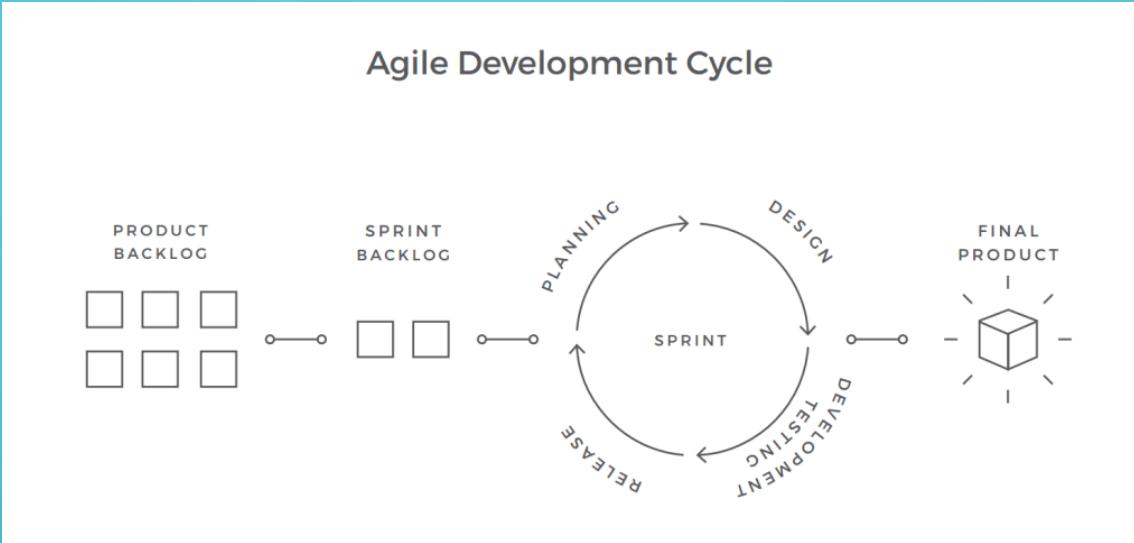




# HOW A DEV TEAM WORKS

# AGILE

- an iterative approach to project management and software development that helps teams deliver value to their customers faster and with fewer headaches
- Team delivers work in small, but consumable, increments instead of betting everything on a "big bang" launch.





## SPRINT

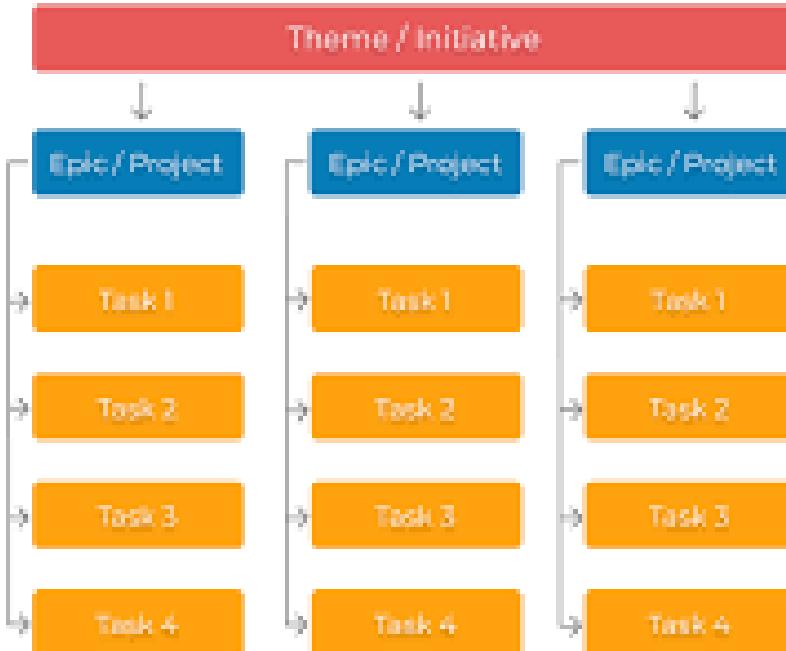
- a short, time-boxed period when a scrum team works to complete a set amount of work
- are at the very heart of scrum and agile methodologies, and getting sprints right will help your agile team ship better software with fewer headaches



## USER STORIES

- short requirements or requests written from the perspective of an end user.

## Agile Tasks Breakdown



## EPICS

- large bodies of work that can be broken down into a number of smaller tasks  
(called stories)



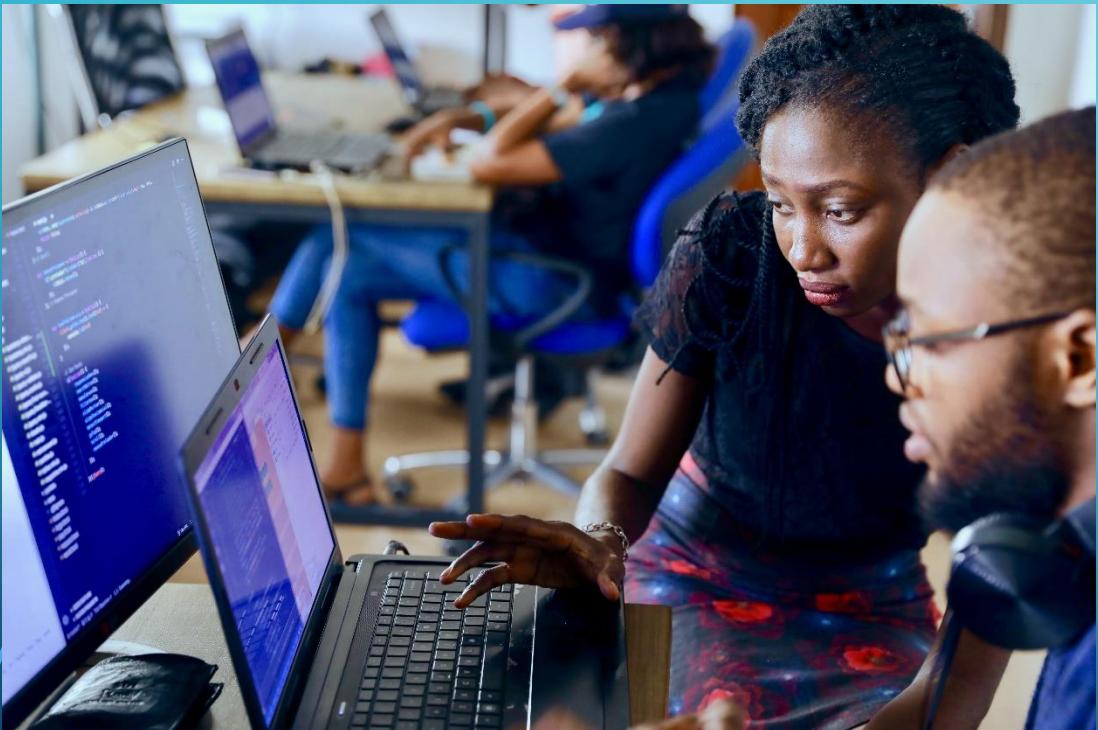
## THE AGILE TEAM

# PRODUCT OWNER



- A member of the Agile Team responsible for defining Stories and prioritizing the Team Backlog to streamline the execution of program priorities while maintaining the conceptual and technical integrity of the Features or components of the team
- the customer proxy responsible for working with Product Management and other stakeholders—including other POs—to define and prioritize stories in the Team Backlog

# SOFTWARE DEVELOPER



- the creative, brainstorming masterminds behind computer programs of all sorts
- **Applications software developer** - responsible for designing computer or mobile applications whose nature of work is largely focused on creating for consumers involving understanding user needs, developing software solutions, monitoring performance and modifying programs as needed
- **Systems software developer** - focused on operating systems-level software, compilers and network distribution software role is geared more toward designing solutions for enterprise organizations including developing software for medical, industrial, military, communications, aerospace, business, scientific and general computing applications

# QUALITY ASSURANCE ENGINEER



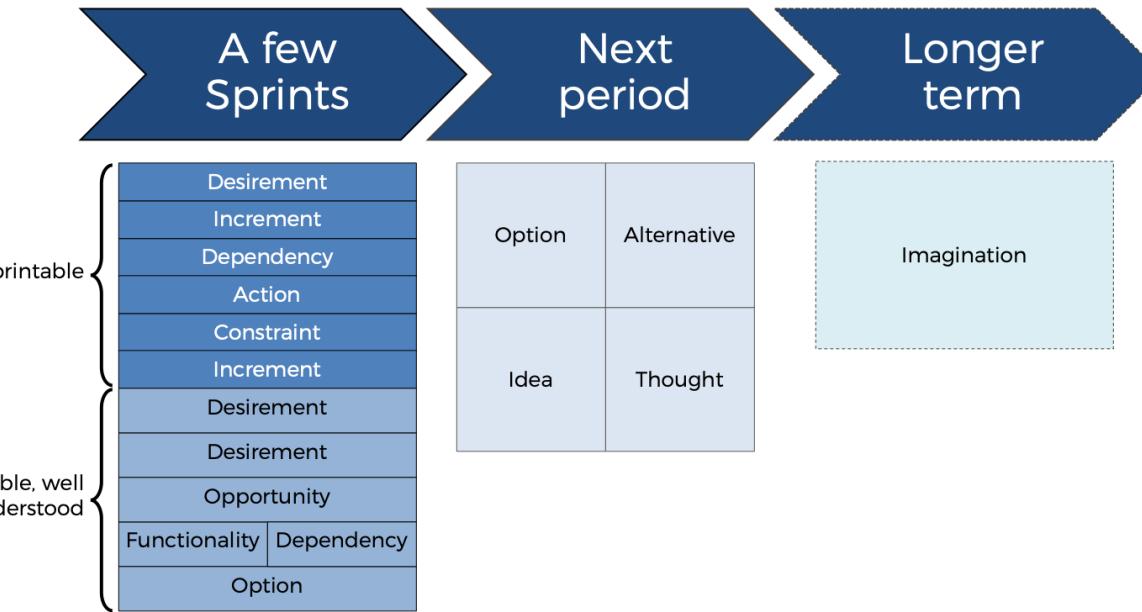
- tracks the development process of a product from start to finish
- oversees production, testing each part to ensure it meets company standards before moving to the next phase
- **Goal:** to create safeguards for products to be market-ready and meet all government regulations before it is released to the public



## KANBAN BOARD

- An agile project management tool designed to help visualize work, limit work-in-progress and maximize efficiency (or flow)
- Uses cards, columns, and continuous improvement to help technology and service teams commit to the right amount of work, and get it done

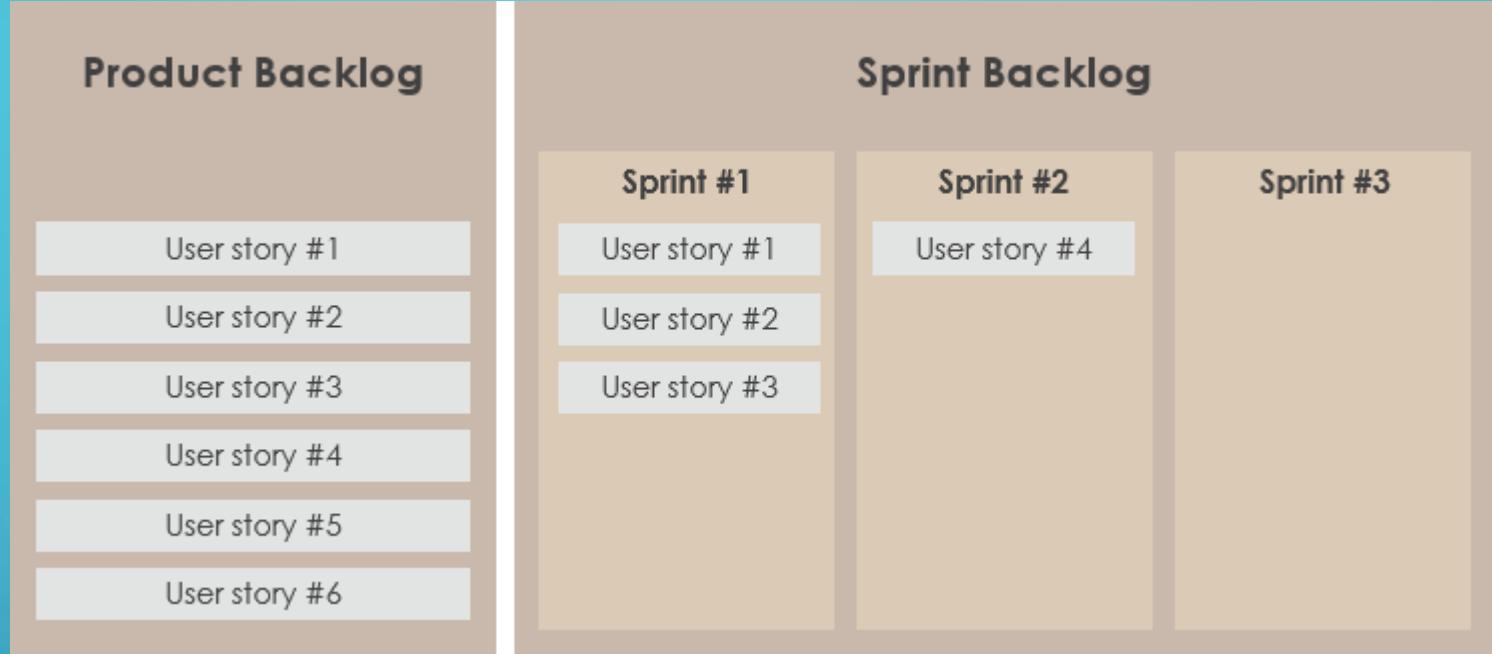
## Product Backlog as a roadmap



(Gunther Verheyen - Ullizee-Inc)

# PRODUCT BACKLOG

- a prioritized list of work for the development team that is derived from the roadmap and its requirements



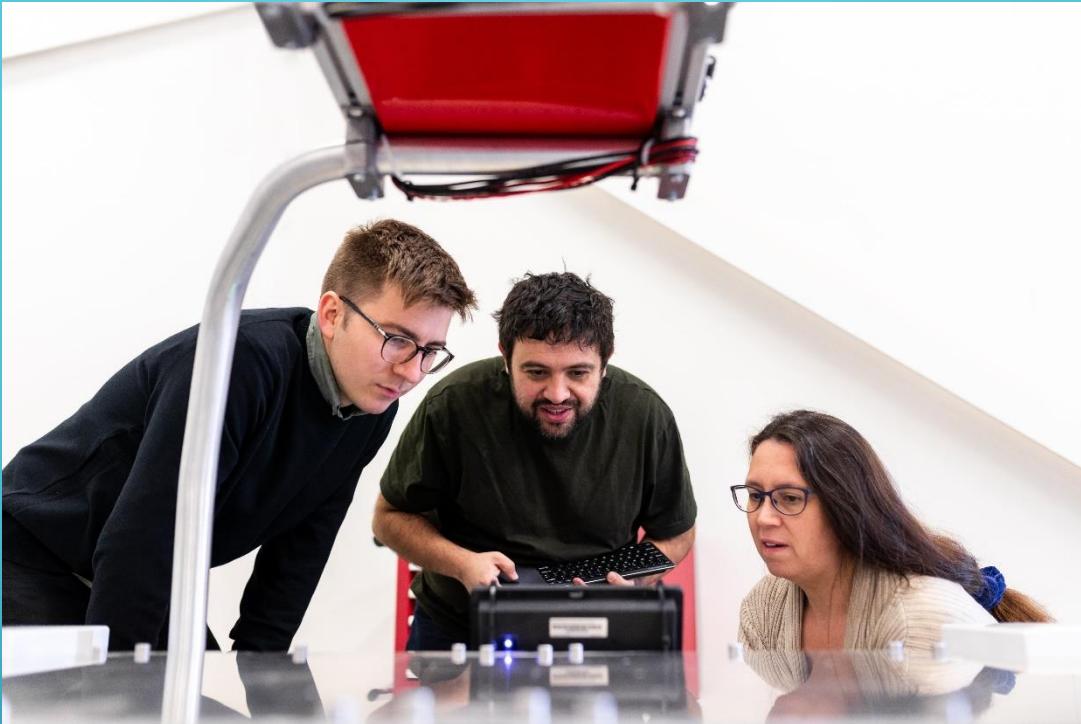
## SPRINT BACKLOG (TO DO)

- like a subset of the product backlog
- comes from the product backlog, but it contains only that item, or those items, that can be completed during each agile sprint

# IN PROGRESS

- Tasks we are doing





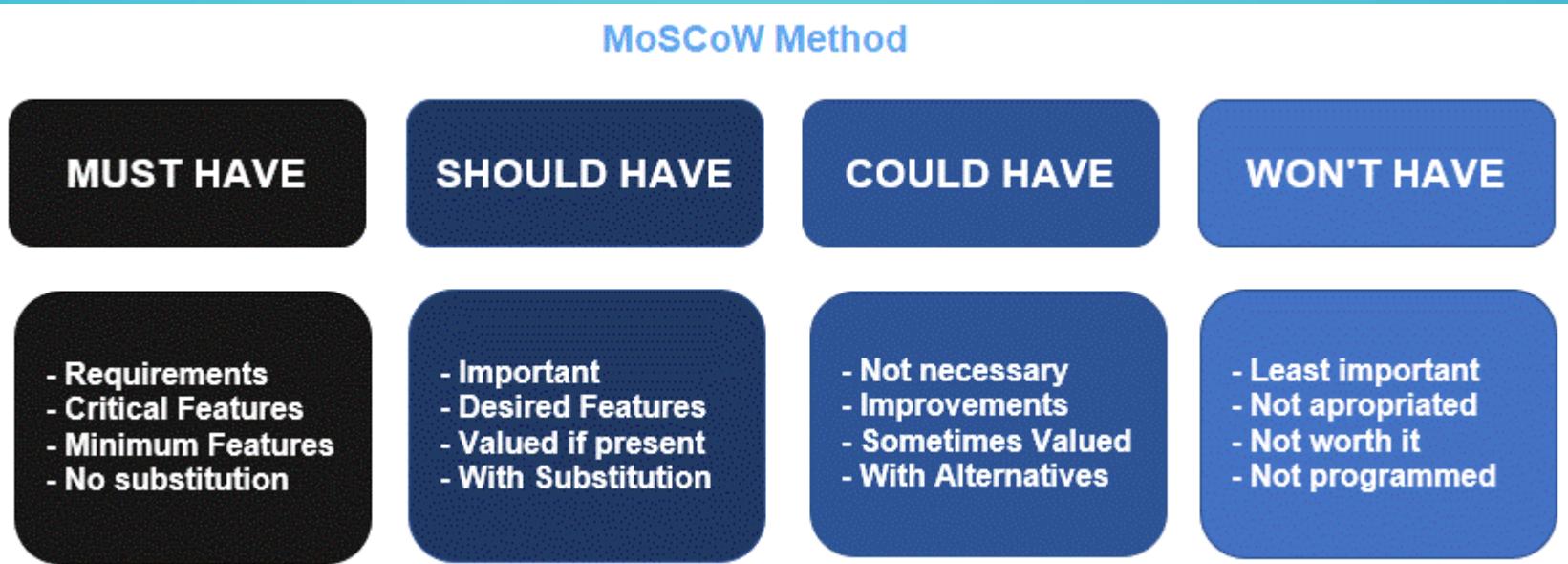
## SOFTWARE TESTING

- a method to check whether the actual software product matches expected requirements and to ensure that software product is Defect free

# DONE

- Completed tasks





## MOSCOW METHOD

- MoSCoW method (a.k.a. MoSCoW prioritization and MoSCoW analysis) is a popular prioritization technique for managing requirements.
- MoSCoW represents four categories of initiatives: must-have, should-have, could-have and won't-have.

# MUST & SHOULD



We use **must** to Express a stronger point of view.  
"We need to ..." "We have to..." The modal **must** also expresses opinion, one person's point of view.

## Examples

- We **must** fasten our seatbelts.
- You **must** stop playing computer games.
- She **must** learn to drive. It will be very useful.
- Banks **must** examine all documents.
- She **must not** disturb me.
- The students **must** study English at least eleven hours a week.

[www.englishgrammarhere.com](http://www.englishgrammarhere.com)

We use **should** to suggest something. "*It is advisable to ...*".  
The modal **should** Express opinion, one person's point of view.

## Examples

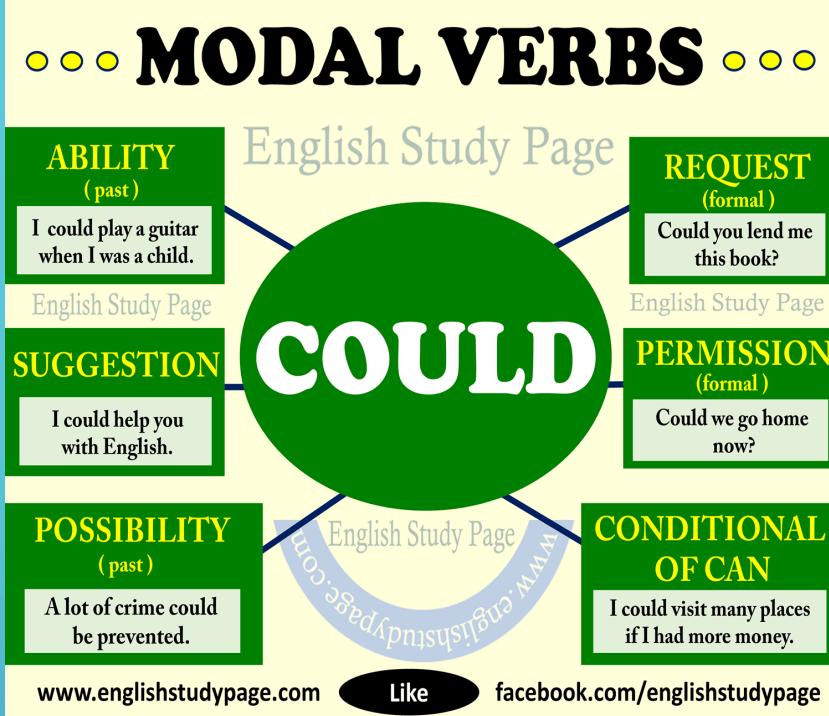
- You **should** apply for that job.
- You have sore throat. You **shouldn't** drink cold water.
- People **should** eat enough fruits in order to be healthy.
- There **should** be a way to do something
- Michale **should** keep out of the sun for two days.

## MUST

- Non-negotiable needs for the project, product or release in question
- Requires the team to complete a mandatory task

## SHOULD

- Essential to the project, product or release but not vital
- Can get scheduled for a future release without impacting the current one
- If left out, the project or product still functions.



# COULD

- Nice to haves, not necessary to the core function of the product
- Compared with should-haves, they have a smaller impact when left out.
- However, the initiatives are often to be first deprioritized if a task in the “should have” or “must have” category ends up larger than expected.



## WILL NOT (WON'T) HAVE (FOR NOW)

- Can manage expectations about what the team will not include in a specific release (or another timeframe you're prioritizing)



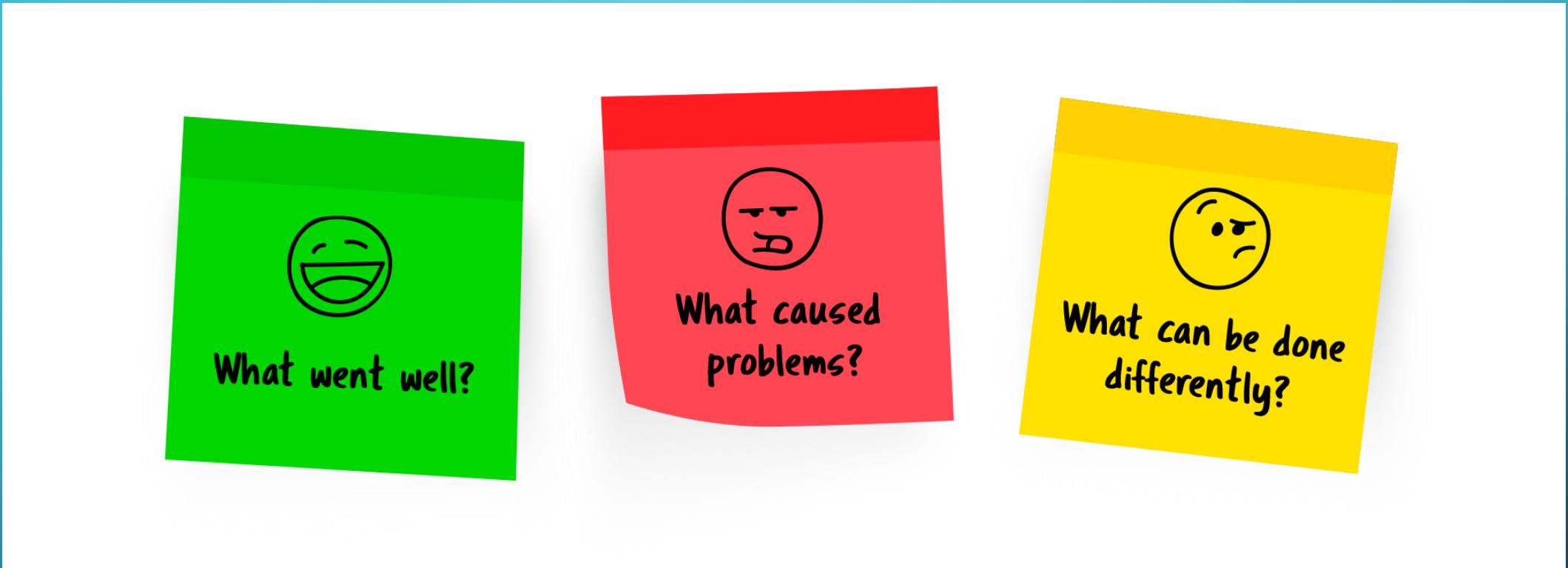
## SPRINT PLANNING

- first day of a new sprint
- an event in the Scrum framework where the team determines the product backlog items they will work on during that sprint and discusses their initial plan for completing those product backlog items



## STAND-UP MEETING

- a short meeting between a team that is held on foot (usually 5 – 15 minutes)
- **Goal:** to go over important tasks that have been finished, are in progress, or are about to be started



## SPRINT RETROSPECTIVE

- a recurring meeting held at the end of a sprint used to discuss what went well during the previous sprint cycle and what can be improved for the next sprint



“I alone know the plans I have for you, plans to bring you prosperity and not disaster, plans to bring about the future you hope for.”

- Jeremiah 29:11 (Good News Translation)

# SOURCES

- <https://kanbanize.com/blog/running-a-better-stand-up-meeting/>
- <https://www.agilealliance.org/glossary/sprint-planning/>
- <https://www.atlassian.com/agile>
- <https://www.atlassian.com/agile/kanban/boards>
- <https://www.atlassian.com/agile/scrum/backlogs>
- <https://www.atlassian.com/agile/scrum/sprints>
- <https://www.digitel.com/kanban/what-is-kanban/>
- <https://www.atlassian.com/agile/project-management/epics-stories-themes>
- <https://www.guru99.com/software-testing-introduction-importance.html>
- <https://www.productplan.com/glossary/moscow-prioritization/>
- <https://www.projectmanager.com/blog/product-backlog-sprint-backlog>
- <https://www.rasmussen.edu/degrees/technology/blog/what-does-software-developer-do/>
- <https://www.scaledagileframework.com/product-owner/>
- <https://www.ziprecruiter.com/e/What-Do-Quality-Assurance-Engineers-Do>